

LIBeLIS

LIBeLIS

JDO

LiDO verleiht Flügel

**Enterprise Information Access
Skalierbare EIA Lösungen**

Alexander v. Zitzewitz
alexander.zitzewitz@libelis.com

✉ info@libelis.com

🌐 www.libelis.com



Visitenkarte LIBeLIS

Enterprise Information Access Software Anbieter

Gründung im Jahr 2000

- Gründer sind Experten in DBMS und J2EE Technologie

Weltweite Aktivitäten

- Niederlassungen in GB, Deutschland und Frankreich
- Distributoren in USA & Kanada
- Erste Kunden in Asien



Finanzierung

- Erste Finanzierungsrunde im November 2002
- Zweite Runde: Juni 2004 – 4 Mio Euro
- Dun & Bradstreet: “Low risk”



Bekennnis zu Java Standards

- Mitglied des Java Community Process (JCP)
- Aktives Mitglied der JDO Expert Group
- JDOCentral Sponsor



LIBeLIS

Der JDO Standard

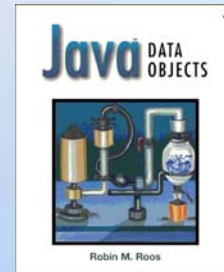
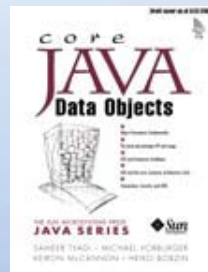
Java Data Objects



- Der Java Standard für Datenzugriff
 - Offizielle Freigabe im März 2002
- Beitragende
 - Sun, IBM, BEA, Oracle, SAP ... & **LIBeLIS**
 - Kein Ersatz, sondern Ergänzung für JDBC und EJB
- Ziele
 - **Transparente** Persistenz
 - Java Code ist frei von technischen Details der Persistenzschicht
 - **Universeller** Datenzugriff
 - RDBMS, ODBMS, Binärdateien, XML, text, Legacy, JCA
 - **Unabhängigkeit** bei der Auslieferung
 - Java Ebene: Client-server J2SE bis hin zu verteilten J2EE Anwendungen



- **Signifikante Verringerung von Projektrisiko und Kosten**
 - **~30% weniger Code**
 - Erheblich geringere Komplexität -> weniger Risiko und Kosten
 - **Risikominimierung**
 - 87% aller gescheiterten Java Projekte scheitern an der Persistenzschicht
 - **Keine Barriere zwischen Datenbanken und Objekten mehr**
 - Modellieren Sie in Java
 - **Verbesserte Herstellerunabhängigkeit**
 - Über 20 JDO Implementierungen, Tendenz steigend...
 - **Datenbank Unabhängigkeit**
 - Austausch der Datenbank ist völlig unproblematisch
 - Benutzen Sie RDBMS, ODBMS, XML, JCA, Dateien, ...
- **Jede Menge Bücher (know-how wiederverwendbar)**



- Flexible Nutzung von JDO Modellen
 - **Nutzung in EJB AppServers**
 - Integration über JCA oder JNDI
 - Verbindungen und Transaktionen laufen über den App Server
 - EJB Anwendungen mit Session Bean Facade Pattern können mit minimalem Aufwand nach JDO portiert werden
 - **JDO in einer Servlet Engine**
 - Oft wird der teure AppServer gar nicht mehr benötigt
 - **Client-Server mit JDO**
 - Cache verbessert Geschwindigkeit
 - Selbes Modell kann auch für Servlet Engine oder App Server benutzt werden.

Alternativen?

- Selbst geschriebene Frameworks
 - **Nicht mehr besonders clever...**
 - Erheblicher Entwicklungsaufwand
 - Erhebliche Wartungskosten
 - Nimmt den Fokus von der eigentlichen Anwendung weg
- Proprietäre Lösungen
 - **Warum wollen Sie sich das antun, wenn es einen Standard gibt?**
 - **Oracle TopLink**
 - TopLink hat häufig den Besitzer gewechselt
 - Oracle ist inzwischen aktives Mitglied der JDO Expert Group
 - Konsequenz ? !
 - **Andere (Cocobase etc...)**
 - Kein relevanter Marktanteil

**Access to Enterprise Information from
Java applications is the biggest
challenge of CIOs today.**

Douglas Clarke, TopLink Product Manager, Oracle, Sept. 2003

**Einverstanden. Aber heutige Anwender
bestehen auf standardisierten
Lösungen.**

Eric Samson, LIBeLIS

Alternativen?

Open Source Frameworks

Etwas smarter als selber schreiben...

- Tool des Jahres Wettbewerb: Castor, Apache OJB, Hibernate, Torque, was kommt als nächstes??
- Proprietär & Guru abhängig (was ist, wenn Gavin King keine Lust mehr hat?)
- Professional Open Source: wird Open Source nun kommerziell?
- Skalierbarkeit, Support, Beständigkeit von API's ??

Entity Beans (EJB 2.0, vergessen wir mal EJB 1.1)

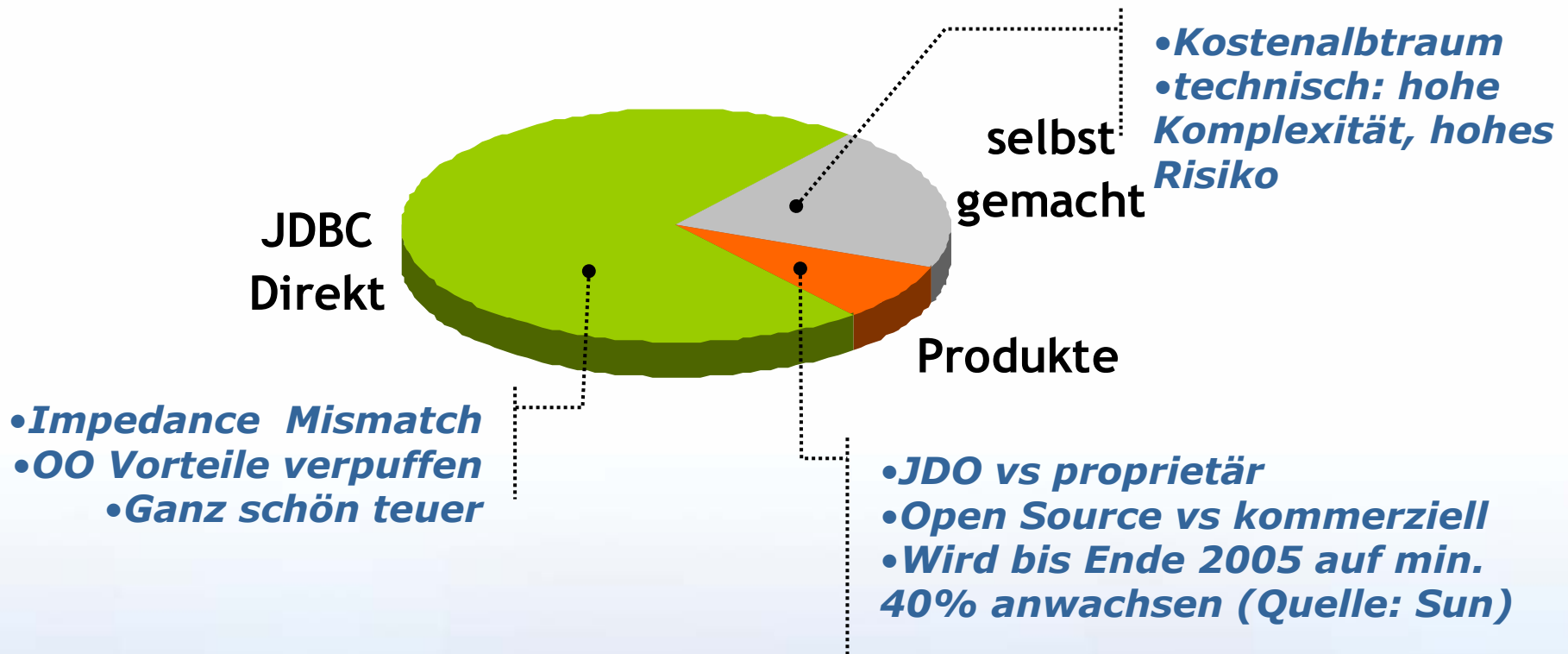
Container Managed Persistence (CMP)

- JDO skaliert erheblich besser (3x – 10x) und ist einfacher anzuwenden
- CMP unterstützt lediglich RDBMS, kein XML etc...
- CMP legt Ihnen erhebliche Einschränkungen bei der Modellierung auf
- In 3 Jahren wird diesem Standard niemand mehr nachweinen

Bean Managed Persistence (CMP)

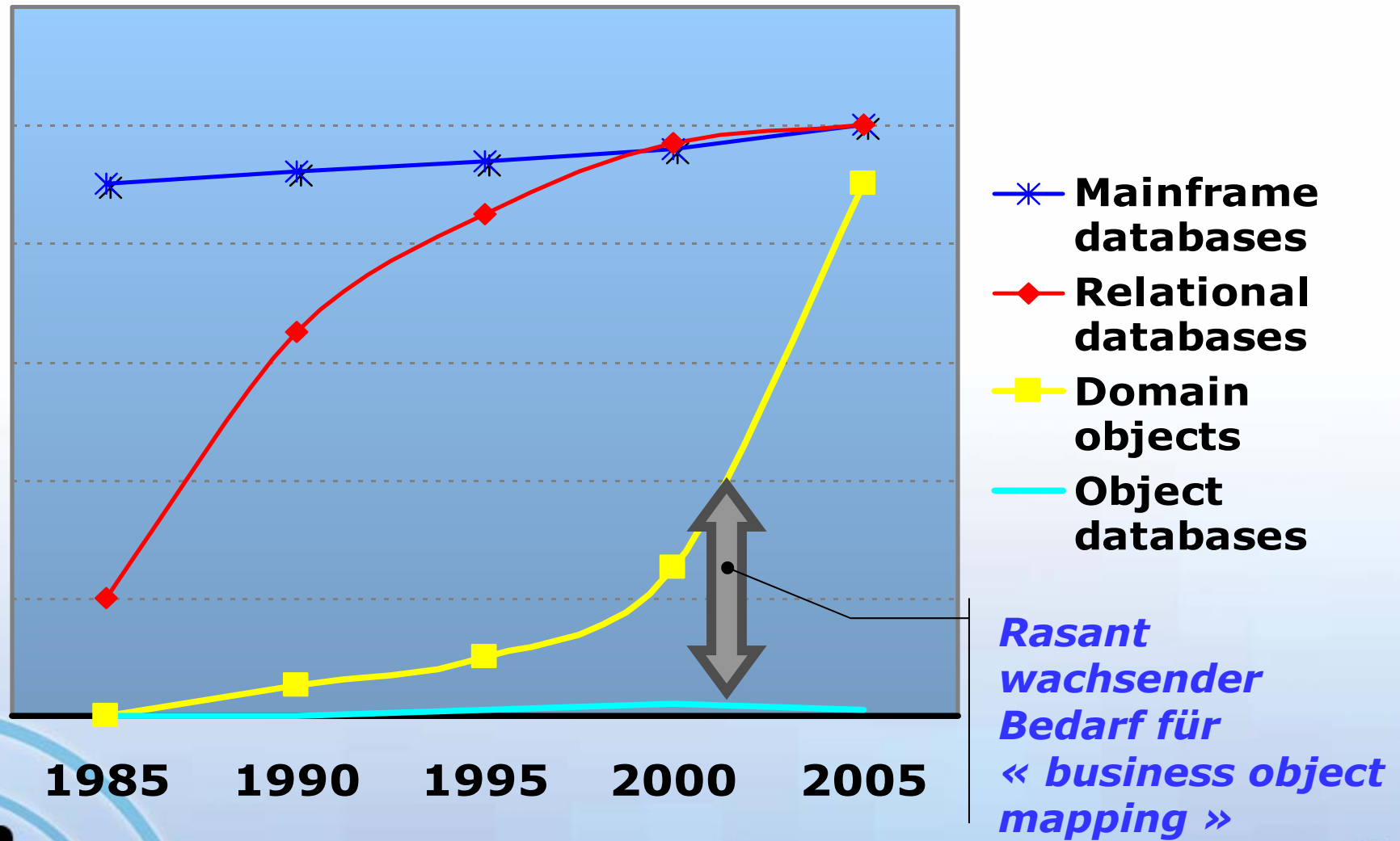
- Wenn Sie nach „Lines of Code“ bezahlt werden möchten ☺

Aktuelle Marktsituation

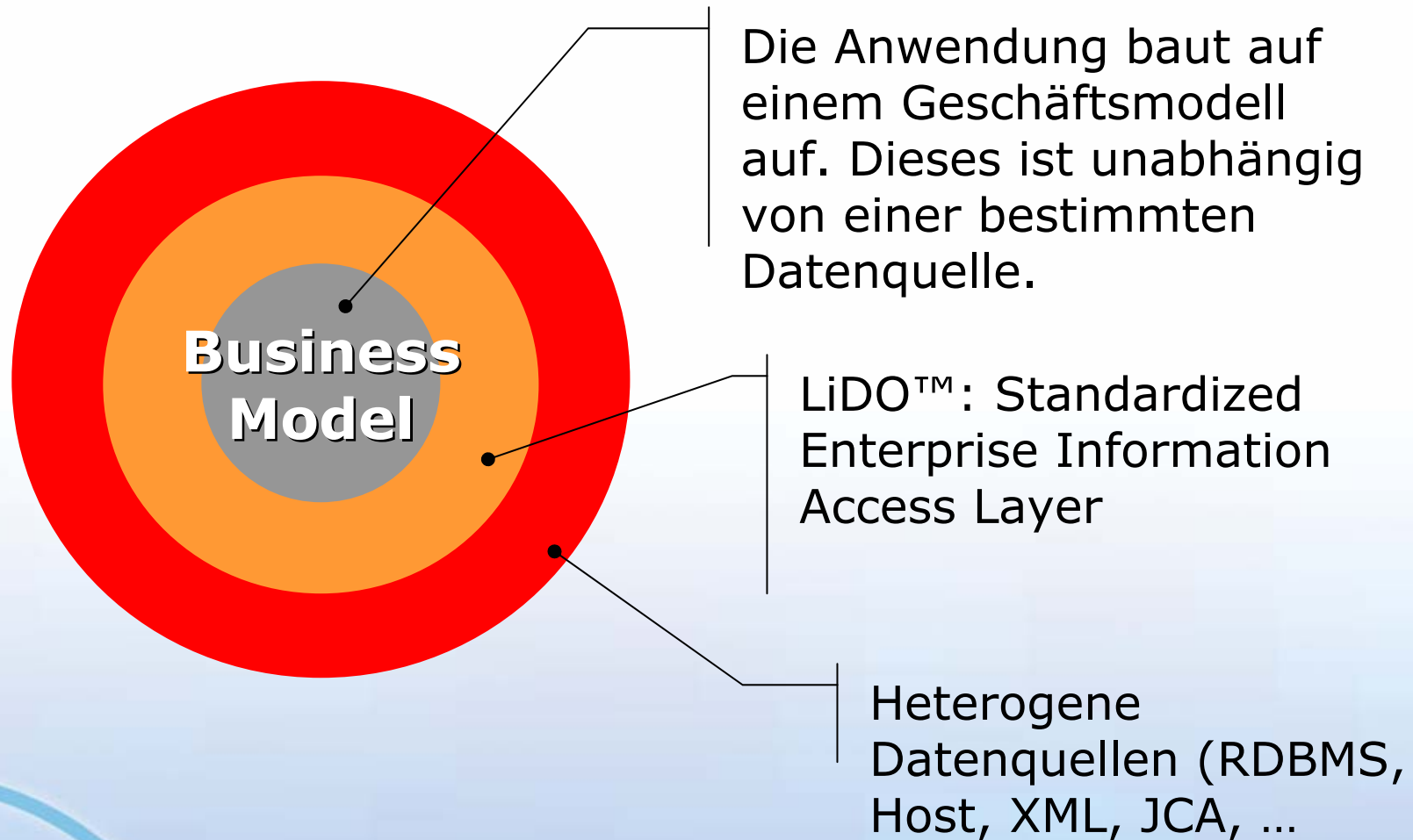


The lack of a robust « information access » product is the main reason for **87% of failed Java projects. The Data Access layer is largely **UNDER-ESTIMATED** !**
Douglas Clarke, Oracle, Sept. 2003

Wachstum für EIA Lösungen



Unsere Vision



Business Model

LiDO™: Standardized Enterprise Information Access Layer

Heterogene Datenquellen (RDBMS, Host, XML, JCA, ...)

JDO, AOP, MDA, SDA are enabling technologies making J2EE effective, with the business at the heart of the IS



LIBeLIS

JDO Programmierung

LiDO aus der Sicht des
Entwicklers



JDO Objektmodell

```
public class Employee extends Person {  
    public String name;  
    private java.util.Date birthday;  
    private int salary;  
    public Company company;  
    public IActivity activity;  
    Vector awards;  
    public void bonus(int more) {  
        salary += more;  
    }  
}
```

Kein javax.jdo
import

Vererbung

Interfaces

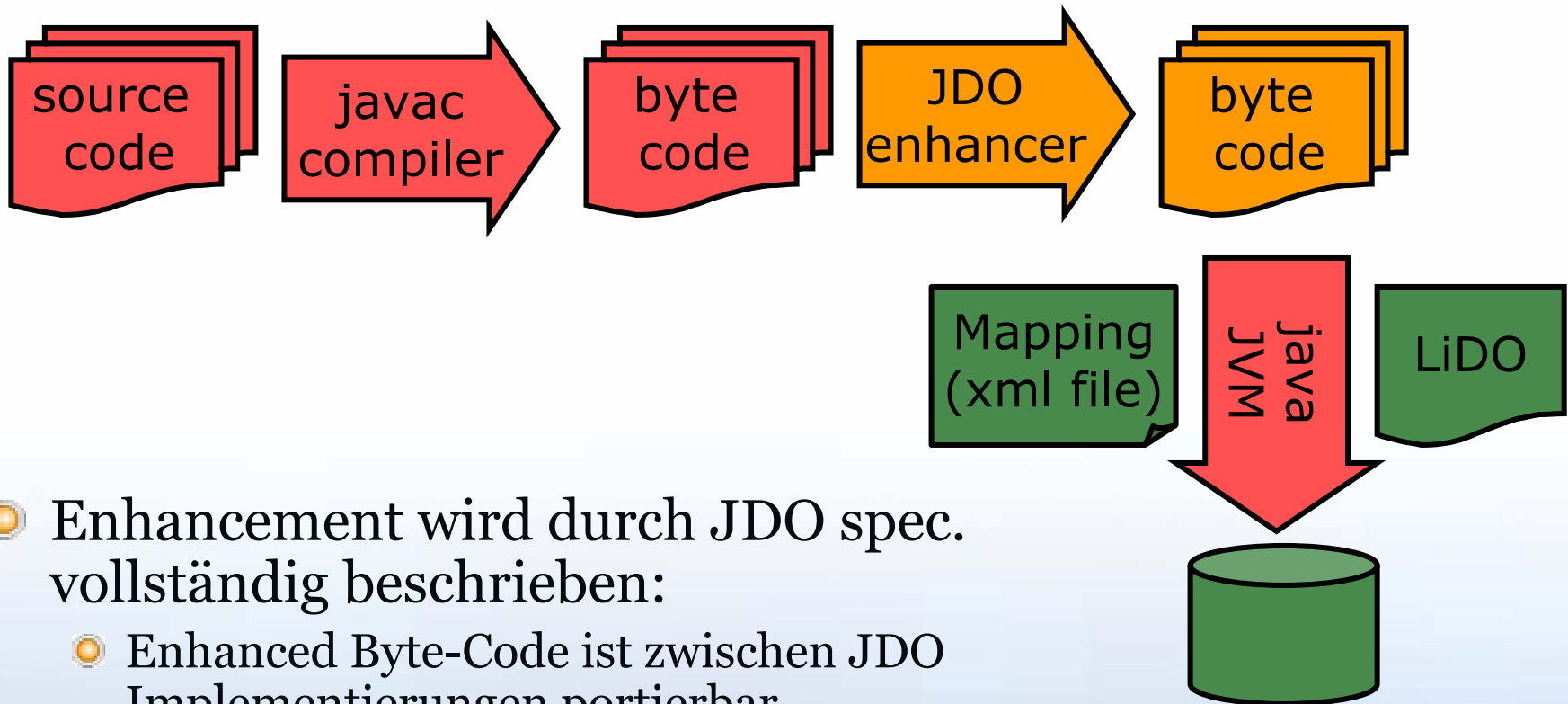
Collections

Keine Accessors
etc.

Das ist eine persistente Klasse !

Wenn Sie es in Java schreiben können, erledigt JDO den Rest für Sie.

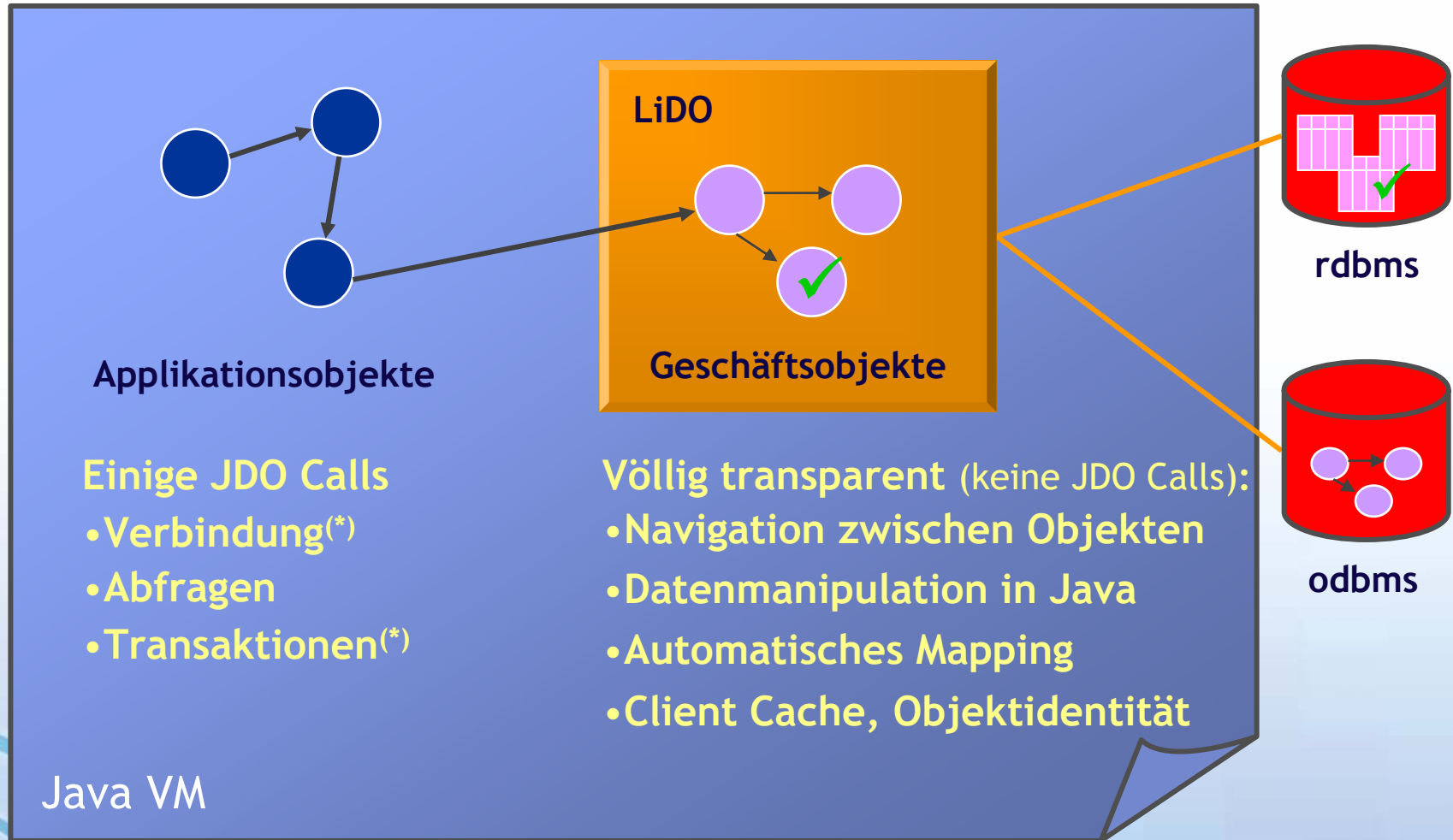
JDO: wie geht das?



- Enhancement wird durch JDO spec. vollständig beschrieben:
 - Enhanced Byte-Code ist zwischen JDO Implementierungen portierbar
 - Enhanced Byte-Code ist unabhängig von der Datenquelle
 - Erlaubt Debugging und Profiling (JPDA)

Erlaubt **völlig** transparente Persistenz

Programmierung mit LiDO



(*) : in einem J2EE Kontext werden Verbindungen und Transaktionen vom AppServer verwaltet !

LIBELIS

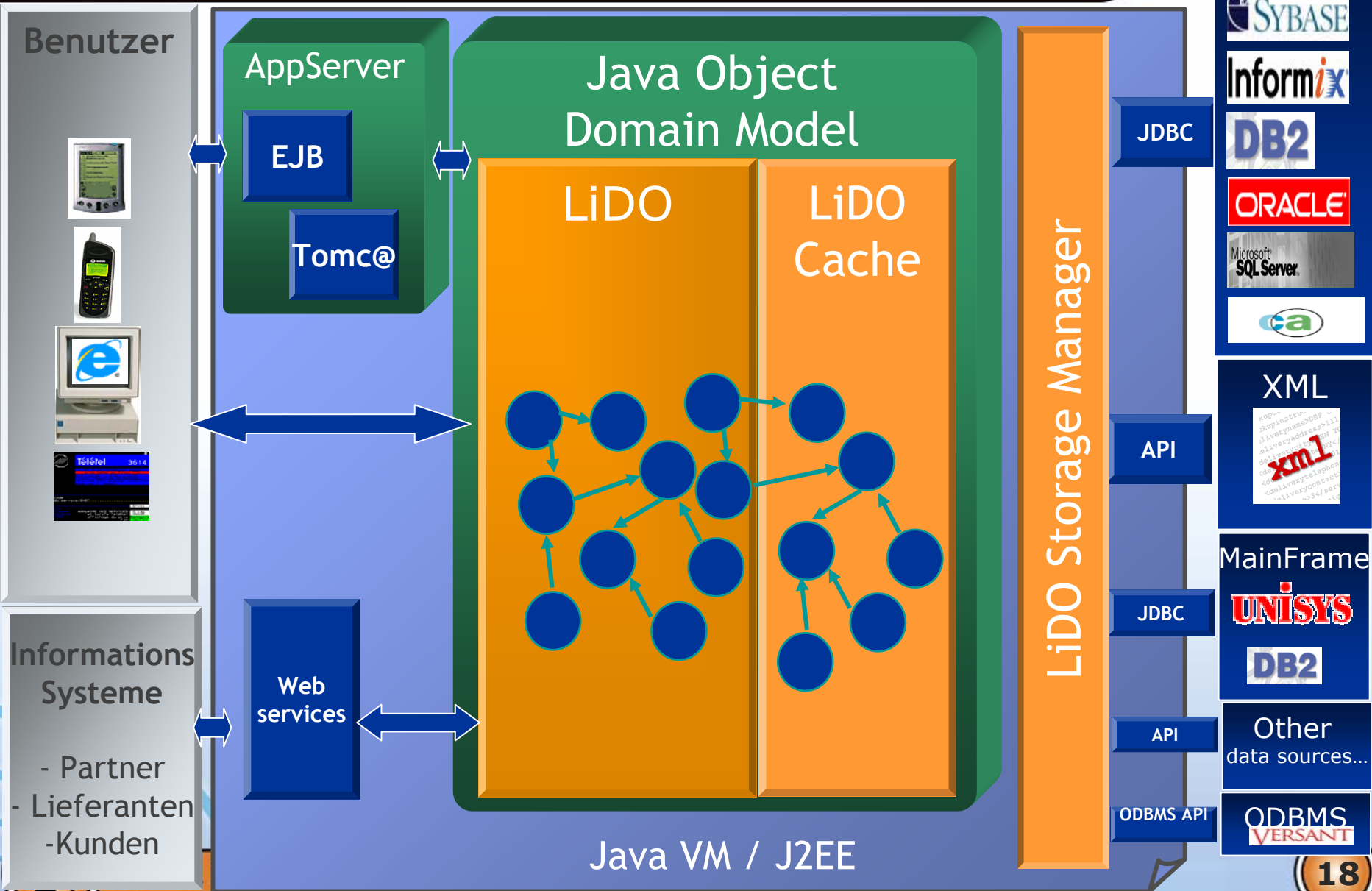
JDO Enterprise Architektur

LiDO aus der Sicht des Architekten

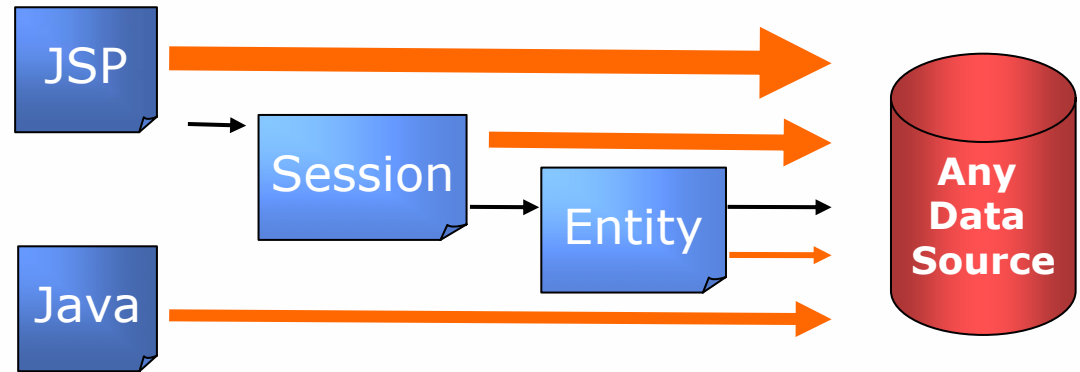
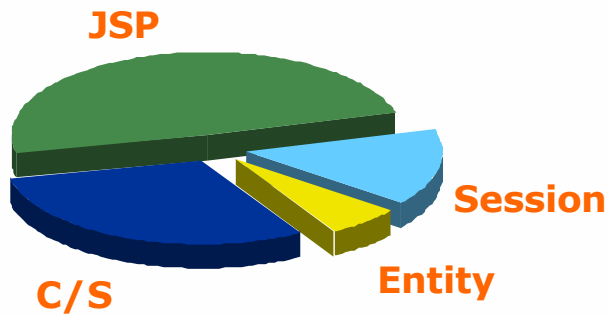
Enterprise Information Access



LiDO JDO Architekturen



Freiheit bei der Architektur



Entity Bean CMP2	JDO-LiDO
Skaliert schlecht	Effiziente Persistenz (leichtgewichtig)
RDBMS und sonst nichts	Universell , jede Datenquelle
Keine OO Modellierung	Java total
Hohe Laufzeitkosten	Infrastruktur unabhängig
Hohe Entwicklungskosten	Reine Java Klassen

JDO wird mehr und mehr als hervorragende Ergänzung für J2EE betrachtet.

LIBELIS

JDO Demo



LiDO Live !

Es läuft nicht nur auf Powerpoint ☺

☉ Mehr als nur JDO

- ☉ Alle JDO Vorteile + LiDO Vorteile
- ☉ Ein API für alle Datenquellen
- ☉ Mehrwertdienste: z.B. Historisierung etc.
- ☉ 3 Jahre auf dem Markt

☉ Exzellente Performance und Skalierbarkeit

- ☉ Erhebliche Einsparungen bei den Betriebskosten
- ☉ Viele erfolgreiche Installationen

☉ ROI < 180 Tage

- ☉ Sparen Sie mindestens 30% Entwicklungsaufwand
- ☉ Mindestens 4 mal mehr Benutzer pro Server
- ☉ **Schnelle Amortisation der Investition**

● **Fiducia IT AG München, Stuttgart**

- 6 Monate Evaluierung mehrerer JDO Tools
- Entscheidung für LiDO wegen Performance und Stabilität
- Java Banking Framework Entwicklung auf Basis von LiDO (>40 Entwickler)

● **Allgemeine Deutsche Direktbank (DiBa)**

- Intranetanwendung für Hypothekendarlehen

● **BNP Cardif Paris**

- 60 Entwickler
- ROI laut Aussage des Kunden in 90 Tagen
- Jährliche Einsparung bei Wartung von 1.000 Manntagen

LIBeLIS Kunden

- **Siemens AG (XML)**
- **XtraMind Technologies GmbH**
- **FinanzIT GmbH**
- **Deltalloyd Versicherungsgruppe**
- **France Telecom**
- **Ericsson (Schweden)**
- **Generali (CH)**
- **Universität Magdeburg**
- **Sanofi Synthelabo**
- **Orange**
- **Sony**
- **HSK (Hong Kong)**
- **Axa**
- **Raytheon (USA)**
- **Rothschild Bank (CH)**



LIBeLIS Partner

- **BEA Systems (BEA Star Partner)**
- **IBM (Reseller Agreement)**
- **ILOG (“Jrules”)**
- **Sun Microsystems**
- **100world AG Nürnberg**
- **cept it Informationstechnologie GmbH
Böblingen**
- **Danet GmbH Weiterstadt**
- **Saxonia Systems AG
Dresden, München, Hamburg, Frankfurt**
- **Syngenio AG (Stuttgart, Bonn, München)**

Neugierig geworden?

**Nutzen Sie unser Starter Package
(bis 30.6.04)**

- **3 Entwicklerlizenzen LiDO Enterprise**
- **1 Tag Coaching vor Ort**
- **1 Testserver Runtime LiDO Enterprise**
- **1 Serverruntime 1 CPU (LiDO Enterprise)**
- **1 Jahr Support (incl. Updates)**

- **Aktionspreis 12.000 € (anstelle von 23.500 €)**